Estimation of Functional Size of a Data Warehouse System using COSMIC FSM Method

Avinash Samuel[1], Anil Kumar Pandey[2], Vivek Kumar Sharma[2]
[1]Deprement of Computer Science & Engineering, Invertis University, Bareilly, India
Avinash_12141@yahoo.co.in
[2]Deprement of Computer Science & Engineering, Invertis University, Bareilly, India
{ Anipandey, Sharma.vivek109}@gmail.com

***Abstract* — It is not easy to measure the functional size of Data Warehouse System. Data Warehouse system is not traditional system and it can be easily measured using FSM (Functional Size Measurement) Method. In this paper we have shown with the help of a case study to measure the functional size of the Data Warehouse System using COSMIC FSMmethod. We will explore the use of COSMIC in sizing Data Warehouse Systems.**

Keywords — Functional Size, Data Warehouse, COSMIC FSM.

I. Introduction

In today's market it has now become a necessity to create a sustainable competitive advantage against competitors by creating a system with which the current operations can be easily tracked, and predict their future strategies. Therefore, one of the trends in the market at the moment is the growing interest in the development of large data warehouses.

It is very hard to predict/estimate the effort and resources required to build a data warehouse system in earlier phases of development. One of the methods proposed by IFPUG (International Function Point User Group) is FPM (Functional Point Analysis). Function Points uses the data movements in the system to measure the end-user requirements. Therefore we can derive an early estimation of the functional size of the system before any code has been written [4]. If development production figures are known, the cost of developing new software can therefore be estimated early enough to make direct comparisons to the cost of buying a software package, or simply using a non-technical solution[4].

The most applied method of measuring functional size is Function Point Analysis (FPA). Another ISO certified method, is COSMIC (Common Software Measurement International Consortium)FSM. In recent history, frameworks have been described of how to measure data warehouse applications with FSM. COSMIC is a better way of measuring the functional size of the software because of the COSMIC's capability to measure the software in different layers and because of the fact that the size of individual functions are not cut off by the maximum size of a function, like in FSM [2].

In this paper, using COSMIC FSM method we measure the functional size of a data warehouse system to understand this concept.

II. Related Work

Functional Size Measurement is a new and emerging field, but many its foundations were laid by Allan Albercht in 1978[10].He was a pioneer or we can say he is the Father of Functional Size Measurement. He proposed IFPUG's FPA which uses Internal logical files, External interface files, External Input, External Output and External Query to measure the functional size of the software. Mark II FPA method or MK II FPA method was proposed by Charles Symons at Nolan Norton in 1984. It uses the Input Data Elements, Entity References and Output Data Elements as the Base Functional Components (BFCs) to measure the functional size of the system. COSMIC FSM method was created by international consortium of industry subject matter experts and academics from 19 countries in the year 1997. It uses data movements (Read, Write, Entry, and Exit) to measure the functional size [9]. The COSMIC FSM method is capable to measure the software having the layered architecture. COSMIC released its guidelines to measure the functional size of the Data Warehouse System, but even with the availability of the guideline it is quite complex to measure the functional size of the data warehouse system. Not much work is done in this field and it is yet too clear how to measure the functional size of the data warehouse system.

III. What Is A Data Warehouse?

A data warehouse is a Subject oriented, integrated, Time variant, Non-volatile collection of data in support of management's decision making process [6].

Note: The data warehouse is always a physically separate store of data transformed from the application data found in the operational environment.

The figure above (Fig 1.) shows the architecture of a data warehouse along with its components. We will now take a look at the components and their functions.

A. Operational data sources

It is the operational system which stores the transactions of the business. It is located outside the data warehouse, as the data warehouse has no control upon the content and the format of the data. The data in these systems is stored in many formats i.e. flat files to hierarchical and relational databases.

DOI: 03.LSCS.2013.3.553

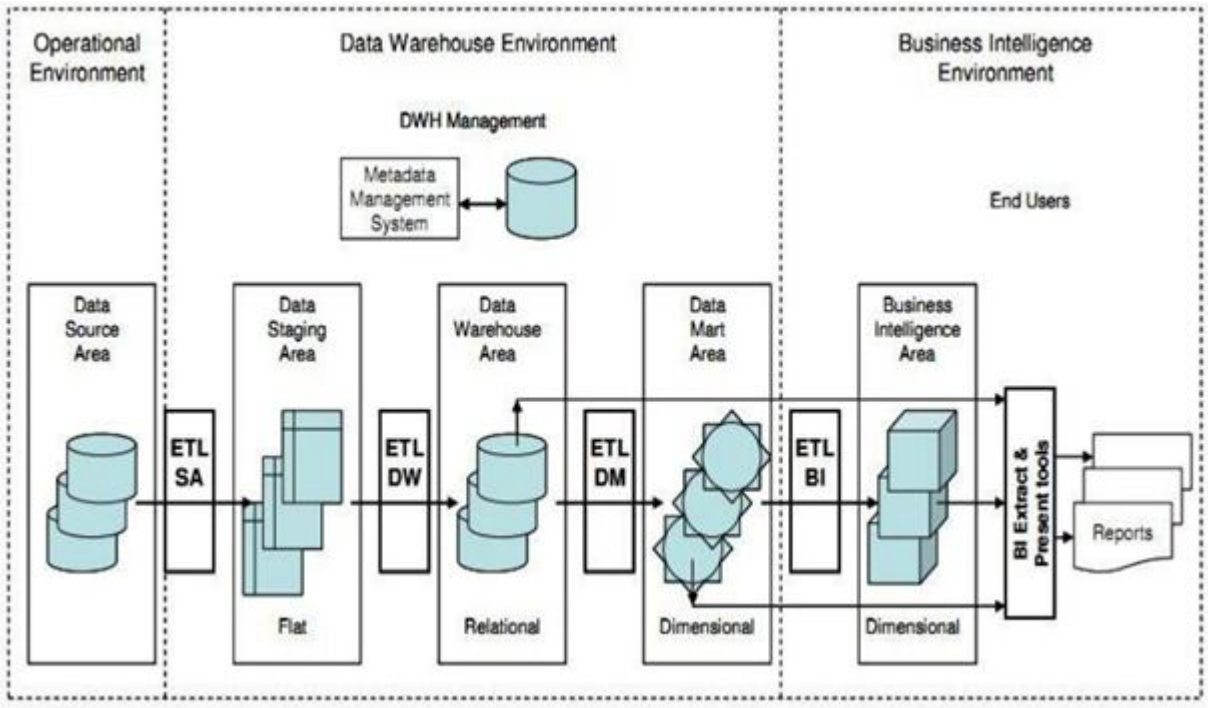


Fig 1: Architecture of Data Warehouse

B. Data Staging Area

Its function is restricted to extracting, cleaning, matching and loading data from multiple source sources. It is explicitly off limits to the end- users, i.e. the end-user has no access or no control over the data staging area. It does not support query or presentation services. A data-cleansing tool is used to process data to resolve name and address misspellings.

C. Extract Transform Load (ETL) Processes

Data-ETL processes are used to extract data from data sources, cleanse the data, perform data transformations, and load the target data warehouse and then again to load the data marts. The ETL processes are also used to generate and maintain a central metadata repository and support the data warehouse [1, 10].

D. Data Warehouse Database

It is a relational data structure that is optimized for distribution. It collects and stores integrated sets of historical, non-volatile data from multiple operational systems and feeds them to one or more data marts [1].

E. Data Marts

Data Marts can be viewed as an extension to the data warehouse. The data comes from the multiple data sources and it is integrated before entering the Data Warehouse System. The data marts contain subject specific information supporting the requirements of the end users in individual business units. Data marts can provide rapid response to end-user requests if most queries are directed to pre-computed, aggregated data stored in the data mart [1].

F. *Metadata management*

Metadata is not the actual data; but rather information that addresses a number of data characteristics such as names and definitions of data elements, where the data comes from, how it is collected, and what transformations it goes through before being stored in the data warehouse [1,2]. Also, meaningful metadata identifies important relationships among data elements that are critical to using and interpreting the data available through the end user query tools.

G. *Business Intelligence (End User functionality)*

Before the end-users can access the data, the data is stored into the business intelligence layer. Here, the data can be visualized as cubes or multidimensional data. There are numerous ways for users to extract the data from the data marts, or from the data warehouse. OLAP tools analyze the data and try to find correlations and meaningful patterns in a fully automated way [1, 2].

IV. Functional Measurement Definitions

A. Functional Size

It is the size of a system/software as viewed from a logical, non-technical point of view. It is more significant to the user than physical or technical size, as for example Lines of Code. This size should be shared between users and developers of the given system [2].

B. *COSMIC FSM Method*

COSMIC is a superset of functional metrics, which provides wider applicability than the IFPUG method. Its key concepts are the possibility of viewing the measured system under different linked layers (different levels of conceptual

DOI: 03.LSCS.2013.3.553

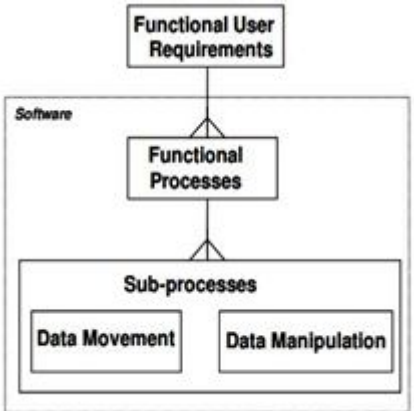
abstraction of the system functions) [2].

V. COSMIC PRINCIPLES

In COSMIC the complete set of requirements only the functional user requirements are measured. The Base Functional Components (BFC's) must be identified within the set of functional user requirements and these BFC's must be measured [4, 9]. The Basic diagram for BFC's is shown in Fig 2.

Base Functional Components in COSMIC are data movement types, which are identified per functional process type. The underlying principles are:

1. Software is activated by input and produces output, or result, that is of use to the user.
2. Software processes parts/pieces of data, which are materialized by data groups, which are a subset of an object of interest (OOI). A data group may consists of one or more data attributes.



The FURs is to be examined and the various functional processes are to be identified. Any of these functional processes consist of a number of sub processes (BFC's), which are called data movements (with included data manipulations).

COSMIC recognizes four kinds of data movement sub processes, Entry, Exit, Read and Write types [4, 9]. The data moments and their data groups are illustrated in Fig 3.

The different data movements and their data groups are as follows:

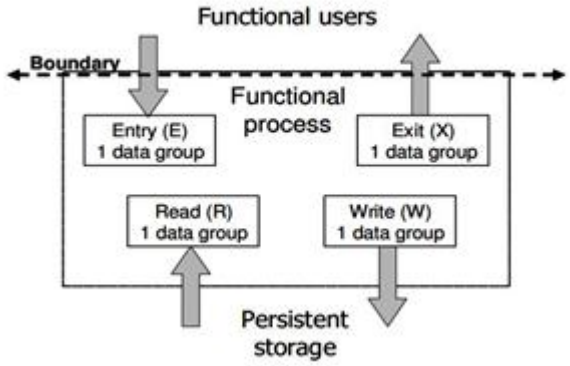


Fig 3: Data Groups in COSMIC FSM Method [3,4]

COSMIC provides the layering concept, which allows the measure to functionally partition the software into different layers, to make sure that all functional processes function on the same level of abstraction. COSMIC also allows the software residing within one layer to be partitioned into peer components, if these components are developed with different technologies, or if they are implemented on different processors[1]. A view of this is presented in Fig 4.

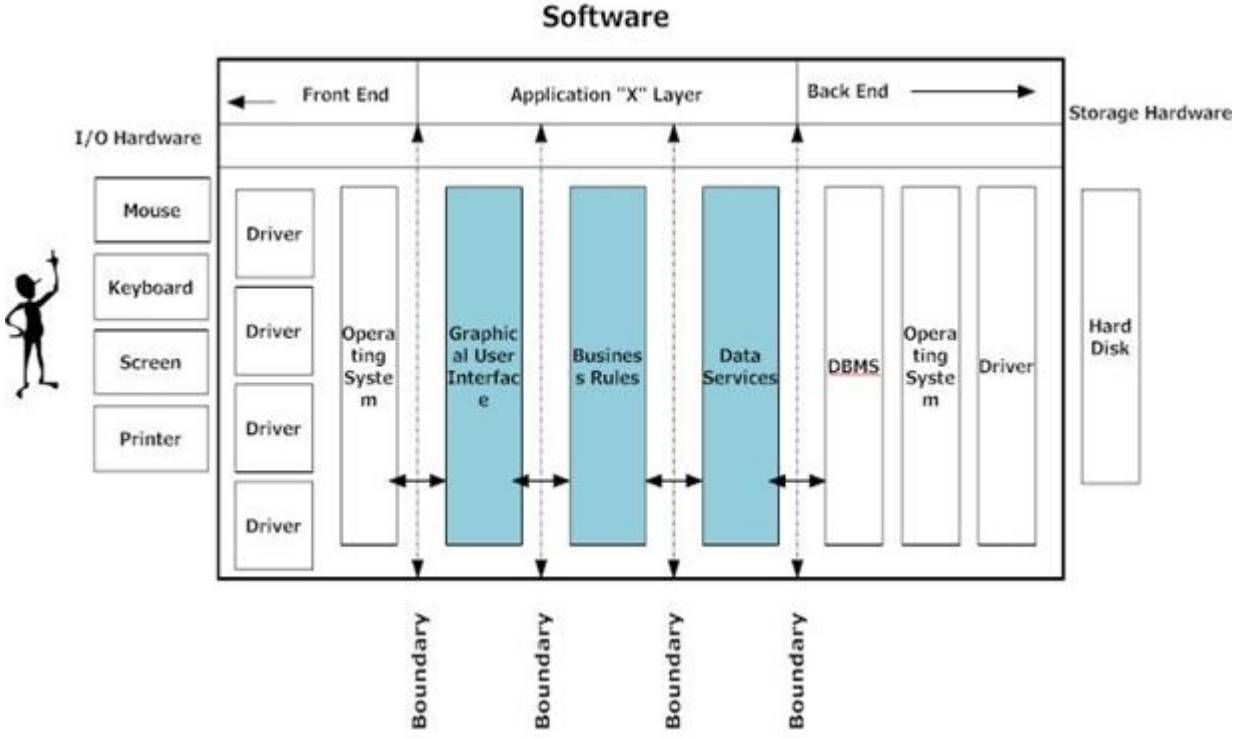


Fig 4: Division of Software into its peer components

DOI: 03.LSCS.2013.3.553

VI. Case Study

To explain the functional size measurement procedure of the Data Warehouse System we take into consideration a data warehouse. The data warehouse that we take for this purpose is the data warehouse of an organization containing its employee's information. As the COSMIC method is capable to measure the functional size of the peer components, we will take a look at the components one by one.

A. Staging Area

The figure shown below (Fig 5) illustrates the initial flow of data from the operational data sources to the staging area.

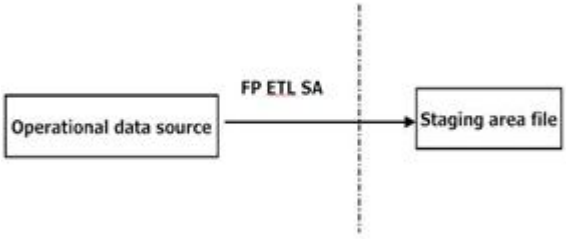


Fig 5: Functional Processes within the Staging Area [1]

Data Movements: For a simple functional process in an ETL (Extraction, Transformation & Loading) Staging Area (SA) tool that must move data about a single OOI-type (where E = Entry, R = Read, W = Write and X = Exit). The Data Movements between the operational data sources, Staging area and metadata is shown in Fig 6.

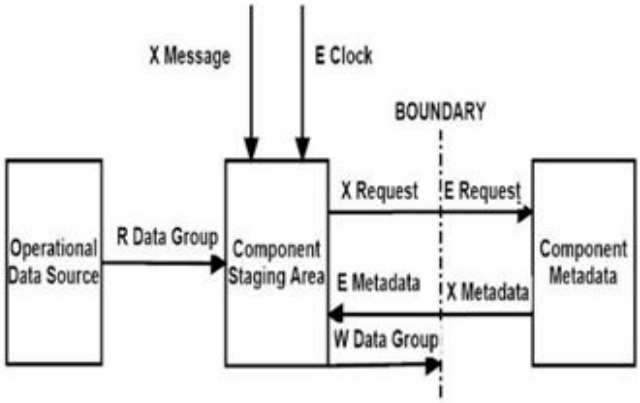


Fig 6: Data Movement within Staging Area Component

Table I illustrates the data movements and their description within the staging area.

Table I: Total Data movements of Staging Area Components

E	To start the functional process (e.g. a clock tick, if a batch process).
X	to the metadata management tool to obtain the transformation rules for this OOI
E	from the metadata management tool with the required metadata
R	of the operational data source
W	of the transformed data to the staging area
X	error/confirmation messages
Total	**6 CFP**

B. Data Warehouse Component

The second component is Data warehouse component. Fig 7 illustrates the flow of data from staging area to data warehouse component.

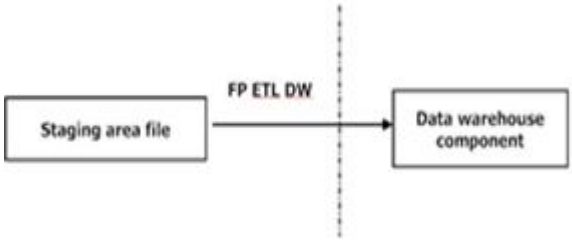


Fig 7: Functional Processes within Data Warehouse Component [1]

Data Movements: A simple functional process of the ETL data warehouse tool that extracts, transforms and loads data describing a single OOI-type would have the data movements as shown in Fig 8.

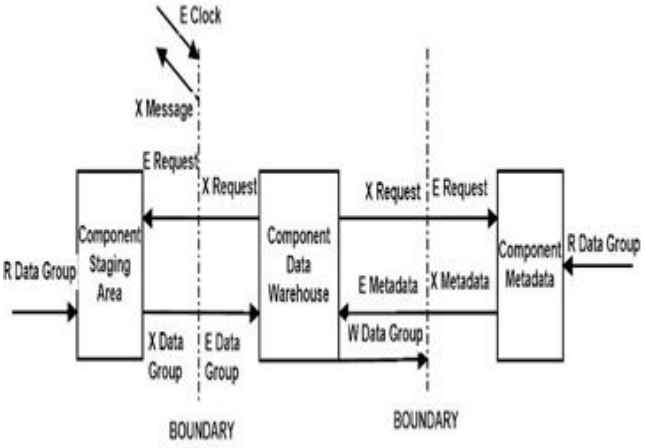


Fig 8: Data Movements within Data Warehouse Components

Table 2 illustrates the data movements and their description within the Data Warehouse Component.

Table II. Total Data movements of Data Warehouse Component

E	To start the functional process (e.g. a clock tick, if a batch process).
X	to the metadata management tool to obtain the transformation rules for this OOI
E	from the metadata management tool with the required metadata
R	of the staging area files
W	of the transformed data to the data warehouse database
X	error/confirmation messages
Total	**6 CFP**

C. Data Mart Component

The Third component is Data mart component. Fig 9 shows the data flow from Data warehouse component to the Data mart Component.

Data Movements: In the ETL data mart tools we find the functional processes that feed the data marts from the data

DOI: 03.LSCS.2013.3.553

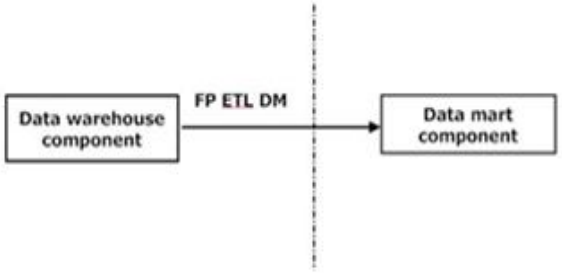


Fig 9: Functional Processes within Data Mart Component [1]

describing the OOI's that are stored in the data warehouse component In the data mart databases, the data will be stored in a dimensional way, thus in star schemas, which shows both 'dimension tables' and 'fact tables' [10]. Fig 10 illustrates the data movements within Data mart component.

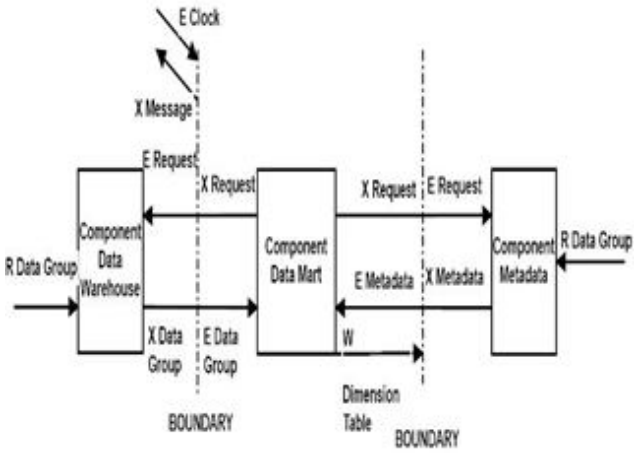


Fig 10: Data Movements within Data Mart Component

Table 3 illustrates the data movements and their description within the Data mart Component.

TABLE III: TOTAL DATA MOVEMENTS OF DATA MART COMPONENT

E	To start the functional process.
R	Read from dimension table of the employee (Employee Output)
R	Read from dimension table of the employee (Employee Personal Information)
R	Read from dimension table of the employee
W	Of the employee's data.
X	error/confirmation messages
Total	**6 CFP**

D. Business Intelligence Components

The fourth component is Business Intelligence component it basically generates reports and provide it to the end user. Fig 11 illustrates the flow of data from the data mart component to the business intelligence component which processes the request of the end users and provides them with generated reports.

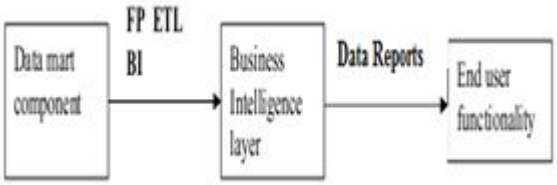


Fig 11: Functional Process within the Business Intelligent Components [1]

Data Movements: There are only three data movements in the system as illustrated in Table 4. The end user sends the enquiry or the query, which is counted as 1 E, the required solution of the query is gathered from the persistent storage 1 R, then the results are returned to the end user in form of some report 1 X.

TABLE IV: TOTAL DATA MOVEMENTS OF BUSINESS INTELLIGENT COMPONENTS

E	to start the enquiry
R	Reads the data from the persistent storage
X	Returns the result
Total	**3 CFP**

E. Metadata Management Component

The metadata administrator have a number of functional processes to his disposal, with which he can create new metadata rules, maintain existing rules or delete metadata rules [10]. User profiles, access privilege files, data processing rules and use statistics can be considered as OOI [1, 7].

Business metadata which are like data dictionaries may contain Data on historical aspects, data on a data owner, etc. [8].

It is good practice in the analysis and design of business application software to check the required stages of the life-cycle of every object of interest (OOI) for which persistent data are held, because each possible transition from one stage to another (in UML terms a 'state transition') should correspond to a functional process. This rule is summarized by the acronym 'CRUD' where C = Create, R = Read, U = Update and D = Delete (sometimes known as 'CRUDL' where L = List). Data about every object of interest must be created, is invariably read, and will usually be updated and deleted, and maybe listed [5].

The Data Movements that are generally encountered within the metadata management component are shown the tables below. Table 5 shows the relationship type i.e. the composite data movements made up of one or more basic data movements.

VII. RESULTS

Table 6 illustrates the different components of the Data Warehouse along with their functional size and the total functional size of the system.

Summing up the functional size of all the peer components we find out that the functional size of our data warehouse comes to about 39 CFP.

DOI: 03.LSCS.2013.3.553

TABLE V: DATA MOVEMENTS OF COMPOSITE RELATIONSHIP TYPES

Composite Relationship type	Data movement types
Creates	1 E, 1 W
Views	1 E, 1 R, 1 X
Lists	1 E, 1 R, 1 X
Changes or Uses	1 E, 1 R, 1 X, 1 W
Deletes	1 E, 1 W
Archives	1 E, 1 W
Updates	1 E, 1 W
Reads	1 R
Total	**18 CFP**

TABLE VI: TOTAL FUNCTIONAL SIZE OF THE DATA WAREHOUSE

Component Name	Calculated Functional Size of Component
Staging Area Component	6 CFP
Data Warehouse Component	6 CFP
Data Mart Component	6 CFP
Business Intelligence Component	3 CFP
Metadata Management Component	18 CFP
Total	**39 CFP**

VIII. CONCLUSION

The sizing of the Data Warehouse will help the Warehouse Administrators to allocate the resources and predict the effort that will be needed. The various data warehouse software components can be separately measured using the COSMIC method due to the layered sizing feature [4].

Using the COSMIC FSM method we have successfully measure the functional size of our data warehouse. Using this information if the developers need to design a similar data warehouse using this information they can easily manage resources, cut down coasts, efficiently manage manpower and complete the project on schedule.

Future work may include developing a new Hybrid FSM method that may include the features of both the COSMIC FSM and MK II FSM method for instance the sensitivity to make small changes to requirements in MK II FSM method is high (detects changes of single data element types and single entity references) where COSMIC FSM's sensitivity to make small changes to requirements is moderate (detects changes to single data-groups) and the smallest feasible enhancement that can be made using the MK II FPA method is 0.26fp where COSMIC FSM can handle the smallest feasible enhancement of 1fp only [11].

Also, new FSM method can be proposed that may include the following characteristics as the current FSM methods are unable to cope with them: Measures corrective maintenance (fixes), Measures perfective maintenance (refactoring for improved performance), Measures algorithmic complexity and Measures reuse of code.

REFERENCES

[1] Van Heeringen, H., Measuring the functional size of a data warehouse application using the COSMIC FFP method, Software Measurement European Forum Conference, Rome, Italy, May 2006.

[2] Santillo, L., "Size & Estimation of data warehouse systems", in FESMA DASMA 2001 conference proceedings, Heidelberg (Germany), May 2001.

[3] "International Software Benchmarking Standards Group database, version 9", January 2008.

[4] "The COSMIC functional size measurement method, version 3.0: Measurement Manual", September 2007.

[5] "The COSMIC functional size measurement method, version 3.0: Guideline for sizing business application software", Version 1.1, May 2008.

[6] Inmon, W.H., "What is a Data Warehouse?" Prism, Volume 1, Number 1, 1995.

[7] Inmon, W.H., "Metadata in a Data in a Data Warehouse: A Statement of Vision", White Paper, Pine Cone Systems, Colorado, December 2005.

[8] Chaudhuri, S. and Dayal, U., "An Overview of Data Warehousing and OLAP Technology", ACM Sigmod record vol. 26 (1), 1997, pp. 65-74.

[9] Sachdeva, S., Meta data architecture for data warehousing, DM Review Magazine, April 1998.

[10] "The COSMIC Method v3.0: Guideline for Sizing Data Warehouse Software", 2009.

[11] "MK II Function Point Analysis Counting Practices Manual v1.3.1", 1998

DOI: 03.LSCS.2013.3.553